

# Assignment 2

● Graded

## Group

Kyle Hoop

Adam Fenjiro

 [View or edit group](#)

## Total Points

54 / 75 pts

### Question 1

#### ERD for movie data

12 / 15 pts

**- 0 pts** Correct

**- 2 pts** Duplicated relation/attribute

**- 2 pts** Incomplete relationship

**- 2 pts** Sequel is a relation, not ISA.

**- 2 pts** No relation between all of the three entities Movie, Studio, and Star.

**- 1 pt** Missing year as a PK for movie.

**- 2 pts** Primary key and weak key denoted improperly

**- 2 pts** Missing primary keys and/or weak keys.

**- 2 pts** Sequel is not an entity or attribute

**- 2 pts** Contract is not an entity

**- 2 pts** Salary is an attribute of Star rather than Star's relationship or missing entirely or an entity

**- 2 pts** Salary is between Movie and Studio?

**- 2 pts** Attribute-less entity

**- 2 pts** Many-to-one relation reversed.

**- 2 pts** Incorrect relation

**- 15 pts** late submission

**- 1 pt** Minor mistake in developing relation.

**- 2 pts** Wrong use of entity

**- 1 pt** Minor mistakes in entity usage and relation.

**Question 2**

- 0 pts Correct

- 30 pts Missing

- 30 pts Late submission

---

#### Entity Set: Department (1)

- 0.5 pts missing PK label

- 0.5 pts missing attribute (should have dept\_name, building budget)

- 1 pt missing

---

#### Entity Set: instructor (1)

- 0.5 pts not a weak entity set

- 0.5 pts missing pk label

✓ - 0.5 pts missing attribute should have id, name salary

---

#### Entity Set: student (1)

- 0.5 pts missing attribute (should have id name credits)

- 0.5 pts missing key label

---

#### Entity Set: course (1)

- 0.5 pts missing pk label

- 0.5 pts missing attributes shou have id, title, credits

---

#### Entity Set: classroom (2)

- 0.5 pts incorrect entity set notation, not a weak entity

- 0.5 pts missing pk label

- 0.5 pts missing attributes

---

#### Entity Set: Section (4)

✓ - 1 pt section is weak entity, double rectangle notation missing

- 0.5 pts incorrcrrect attributes (should have sec\_id, semester, year, capacity)

- 0.5 pts missing weak key label

✓ - 1 pt missing attributes hould have sec\_id, semester, year, capacity)

- 3 pts missing

---

## Relationship: dept - course (1)

- 0.5 pts incorrect multiplicity, it should be a one to many relationship (arrow to dept)

✓ - 1 pt incorrect notation or missing

- 0 pts Click here to replace this description.

---

## Relationship: dept - instructor (1)

- 0.5 pts works for should be relationship between dept and instructor not an attribute of instructor

- 1 pt do not directly link instructor entity set and dept

- 0.5 pts arrow to dept not instructor

✓ - 1 pt missing

---

## Relationship: dept - student (1)

- 0.5 pts major should be relationship between dept and student not an attribute of student

- 1 pt missing major relationship

- 0 pts Click here to replace this description.

---

## Relationship: instructor - section (2)

- 1 pt Teaching should be relationship between instructor and section

✓ - 0.5 pts it should be one to many relationship (arrow to instructor)

---

## Relationship: course - section (2)

✓ - 0.5 pts it should be an identifying relationship, section is a weak entity set

- 2 pts missing section of relationship between section and course

---

## Relationship: student - section (2)

- 1 pt missing

- 0.5 pts it should be many to many relationship

✓ - 0.5 pts missing attribute grade

---

## Relationship: section - classroom (2)

- 0.5 pts incorrect multiplicity it should be many to one, (Arrow point to classroom)

- 0.5 pts incorrect shape

✓ - 1 pt missing

---

## Relationship: course - course (preq) (1)

✓ - 1 pt preq is a relationship from course to itself

- 0.5 pts its not identifying relationship

---

- 2 pts see comment

💬 - 1 pt incorrect notation for connecting attribtue and entity set

### Question 3

From textbook

21 / 30 pts

#### 3.1 Exercise 4.1.3

6 / 10 pts

- 0 pts Correct
- 1 pt missing primary key label
- 1 pt missing attributes
- 1 pt minor I error
- 2 pts minor II error
- 4 pts major error
- 10 pts missing
- 10 pts Late submission

✓ - 2 pts Missing uniform relation and color entity.

- 1 pt plays\_on is not an identifying relationship.
- 1 pt captain incorrectly uses ISA instead of being a relationship.

💬 - 2 pts No ISA.

#### 3.2 Exercise 4.2.1

6 / 10 pts

- 0 pts Correct

✓ - 2 pts Did not mention the redundancy of AcctSets existing rather than simply using a many-to-many relation between accounts and customers and getting rid of AcctSets entirely

✓ - 2 pts Did not mention the over-complication of having an address entity/relationship when an address attribute on customer would suffice instead, or a multivalued attribute if necessary

- 10 pts Missing

- 5 pts Wrong diagram analyzed

- 10 pts Late submission

1 Which entities? Which relationships?

3.3

Exercise 4.4.4 a)

9 / 10 pts

- 0 pts Correct

- 1 pt The weak key is not indicated by a dotted underline

- 1 pt The primary key is not indicated by an underline

- 1 pt Wrong question

✓ - 1 pt The supporting relation is not a diamond inside another diamond

- 1 pt The weak-entity set entity is not a box inside another box

- 10 pts Missing

- 6 pts Not an ER diagram

- 2 pts dept is not a weak entity set

No questions assigned to the following page.

## CS3425 Assignment 2

(75 points)

### Goal

Design relational database from user requirement

1. Get familiar with ER diagram notation
2. Use ER diagram to design conceptual model for real app
3. Convert ER diagram to relational schema
4. Communicate with user to understand and refine user requirement

### Important notes:

1. Please use the notation that we covered in class. You will get 0 if you use other notations.
2. The computer in the Linux labs has software "dia" supports ER diagram. You may download and install dia on your own computer from here: <http://dia-installer.de/download/index.html>

If you are use Mac and have trouble to start Dia from the launchpad, try to change the setting:

From Applications: Dia->Show Package Contents -> Contents -> Resources -> bin -> and edit the 'dia' file to include the line " export DISPLAY=:0 " outside the "if" statement. See the redline below.

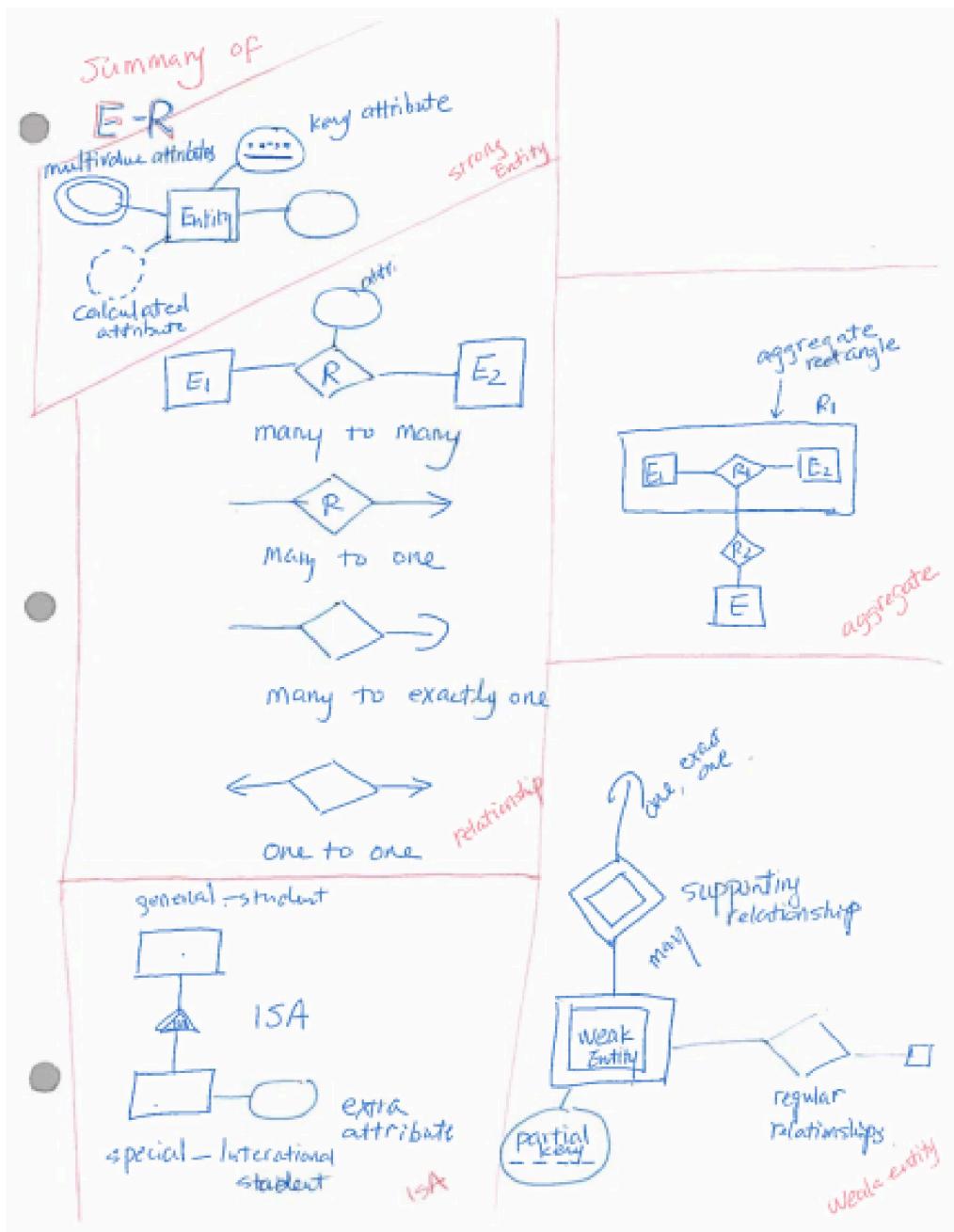
```
cd /Applications/Dia.app/Contents/Resources/bin
```

```
vi dia
```

```
# Check for X11
if [[ "" == $DISPLAY ]]; then
    export DISPLAY=:0
fi
export DISPLAY=:0
```

3. The graph in **dia** can be exported as PNG format, which can be inserted into Word or other software that you use to create your report in PDF.
4. Three are total of 3 problems. Please start a new page for each question. When you upload your submission to grade scope, please carefully choose the correct pages for each question.

No questions assigned to the following page.



Question assigned to the following page: [1](#)

### **1. (15 points) Design ER model for movie data**

**Design ER model for the following requirement. If there is any not well-defined requirement, please write down your own reasonable assumption in your report. You may also see book Page**

**In these data base, we would like to store information about movies, stars and studios.**

- For stars, we would like to keep track of their names and addresses.
- For studios, we would like to keep track of studios' names and addresses
- Movies have title, year they were made, length, and type.

**Besides the above basic information, we would like to store data about**

- which stars acted in which movies.

**We would like to keep track of the contacts between studio and star.**

- A studio has contacted a particular star to act in a particular movie for a certain payment (salary).

**We would also like to keep track of**

- which movie is the sequel of the other movie. For example: T2 is the sequel of Original Terminator, T3 is sequel to T2, Terminator Salvation is sequel to T3.

**stars(name, address)**

PK: name

**studios(studio\_name, address)**

PK: studio\_name

**movies(title, year, length, type)**

PK: title

FK: titles refs movies(titles)

**acted(name, title)**

PK: name, title

FK1: name refs stars(name)

FK2: title refs movies(title)

**works\_for(name, studio\_name, salary)**

PK: name, studio\_name

FK1: name refs stars(name)

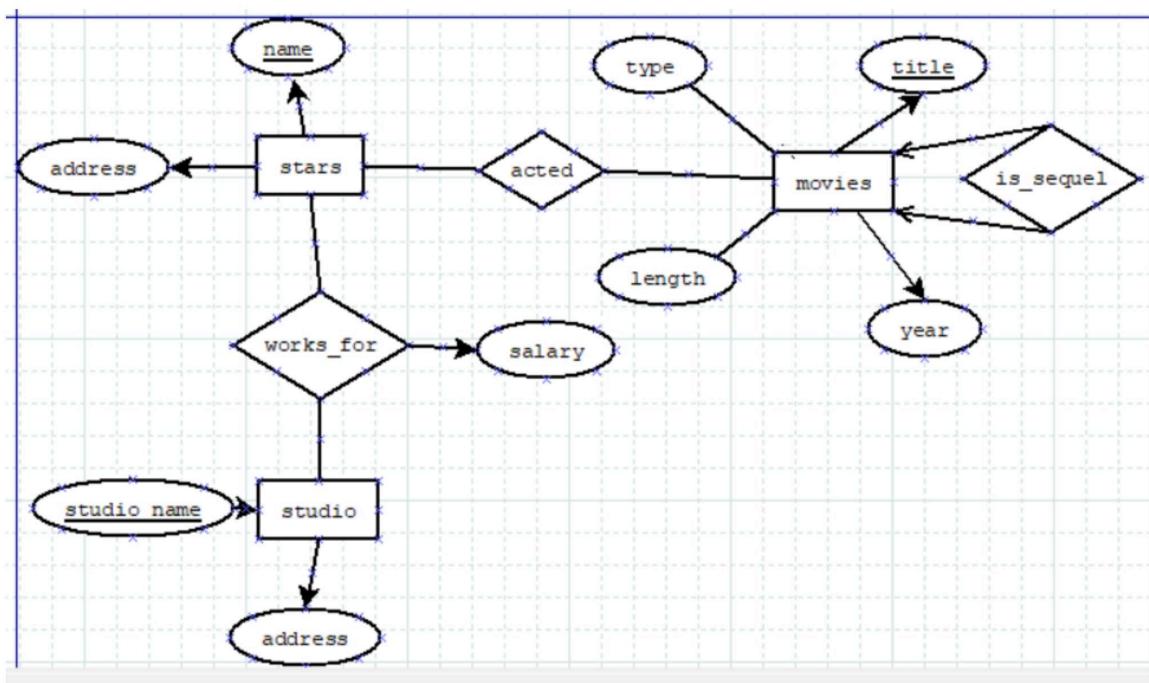
FK2: studio\_name ref studio(studio\_name)

**is\_sequel(title)**

PK: title

FK: title refs movies(titles)

Question assigned to the following page: [1](#)



Question assigned to the following page: [2](#)

## 2. (30 points) Design ER model for the university

The major characteristics of the university:

- The university is organized into departments. Each department is identified by a unique name (dept\_name), is located in a particular building, and has a budget.
- Each department has a list of courses it offers. Each course has associated with it a course\_id, title, and credits, and may also have associated prerequisites.
- Instructors are identified by their unique IDs. Each instructor has name, associated department (dept\_name), and salary.
- Students are identified by their unique IDs. Each student has a name, an associated major department (dept\_name), and tot\_cred (total credit hours the student earned thus far).
- The university maintains a list of classrooms, specifying the name of the building, room\_number, and room capacity.
- Each class may have been offered as multiple sections in certain semesters. The university maintains a list of all sections that have been offered. Each section is identified by a sec\_id for the same course in every semester. Each section has its own capacity, building and room\_number.
- The department has a list of teaching assignment specifying, for each instructor, the sections the instructors teach.
- The university has a list of all student course registrations, specifying, for each student, the courses and the associated sections that the student has taken (registered for) and the grade they got.

If there is any not well-defined requirement, please write down your own reasonable assumption in your report.

Question assigned to the following page: [2](#)

Department(dept\_name, budget)

PK:dept\_name

Course(course\_id, title, credits, prerequisites)

PK:course\_id

Instructors(instructor\_id, instructor\_name)

PK:instructor\_id

Students(student\_id, student\_name, major, tot\_cred)

PK:student\_id

Classrooms(classroom\_id, room\_capacity)

PK:classroom\_id

Create a building entry that connect to department, classroom and section and has a name and a location

building(bName, location)

PK: bName

Subclass of course

Section(sec\_id, sec\_capacity)

PK:sec\_id

FK: sec\_id refs section(sec\_id)

Connects instructors to sections

Teaching\_assignment(instructor\_id, sec\_id)

PK:instructor\_id, sec\_id

FK1:instructor\_id refs instructor(instructor\_id)

FK2: sec\_id refs section(sec\_id)

Connects student and course

Register\_in(student\_id, course\_id, sec\_id, grade)

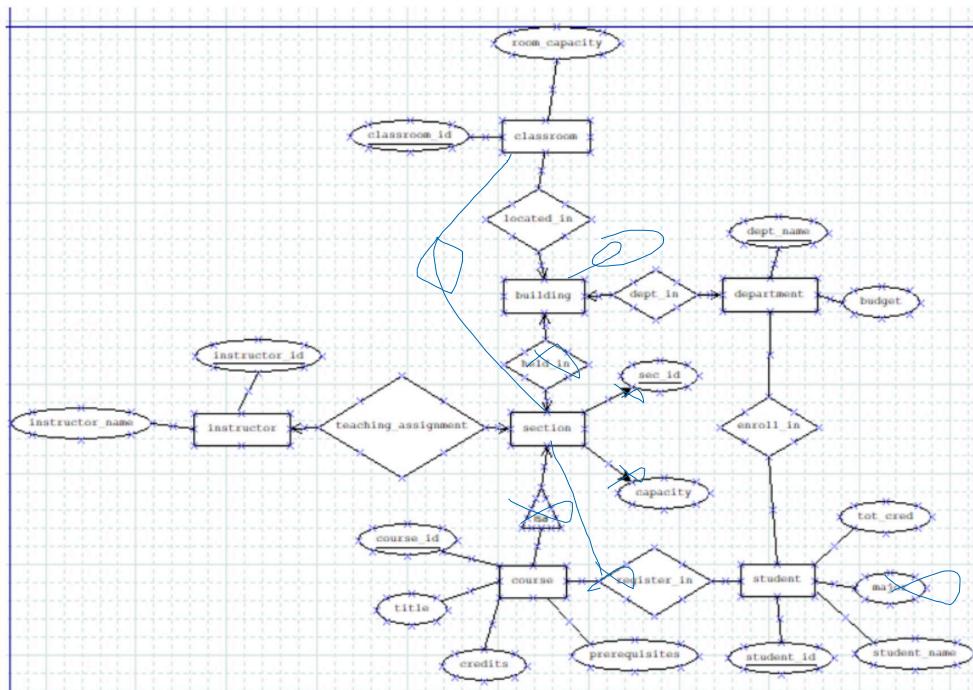
PK:student\_id, course\_id

FK1:student id refs student(student\_id)

FK2:course\_id refs course(course\_id)

Question assigned to the following page: [2](#)

PK:registration\_id  
 Connects instructors and departments  
 teaches\_for(instructor\_id)  
 PK:dept\_name  
 FK2: instructor\_id refs instructor(instructor\_id)  
 Connects student and department  
 enroll\_in(student\_id)  
 PK: dept\_name  
 FK:student\_id refs student(student\_id)  
 Connect building and department  
 dept\_in(dept\_name, bName)  
 PK:dept\_name  
 FK1:dept\_name refs department(dept\_name)  
 FK2:bName refs building(bName)  
 Connect building and section  
 held\_in(sec\_id, bName)  
 PK:sec\_id  
 FK1:sec\_id refs section(sec\_id)  
 FK2:bName refs building(bName)  
 Connects building and classroom  
 located\_in(classroom\_id, room\_number)  
 PK: bName  
 FK: classroom\_id refs classroom(classroom\_id)



Question assigned to the following page: [3.1](#)

### **3. (30 points) Exercises from the textbook**

#### **3.1. (10 points) Exercise 4.1.3 on page 139**

**Exercise 4.1.3:** Give an E/R diagram for a database recording information about teams, players, and their fans, including:

- 1. For each team, its name, its players, its team captain (one of its players), and the colors of its uniform.**
- 2. For each player, his/her name.**
- 3. For each fan, his/her name, favorite teams, favorite players, and favorite color.**

team(tName, color)

PK:tName

player(pName)

PK:pName

team\_captain(pName)

PK:pName

FK: pName refs player(pName)

fan(fName, fColor)

PK:pName

plays\_on(pName)

PK:tName

FK:pName refs player(pName)

favorite\_team(fName)

PK:tName

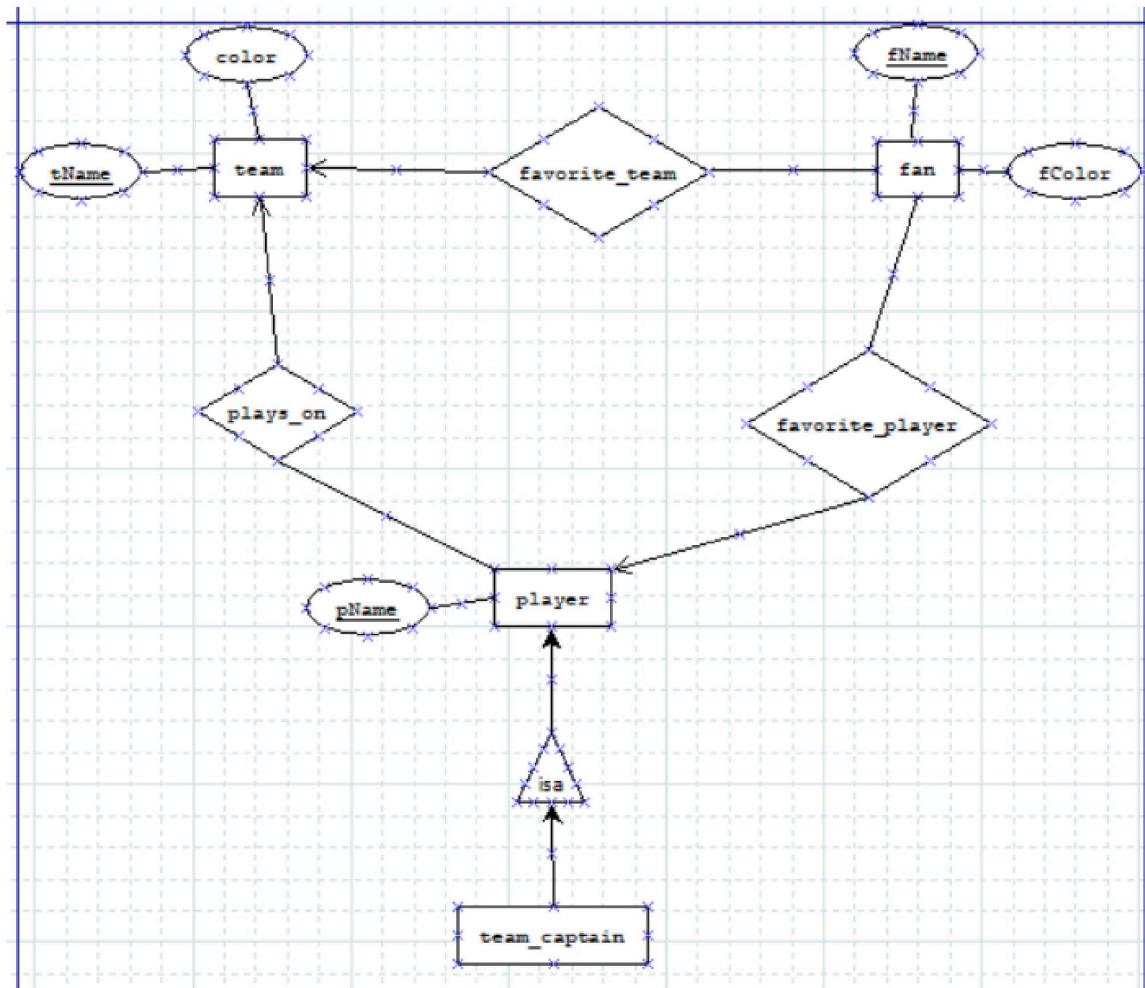
FK: fName refs fan(fName)

favorite\_player(fName)

PK:pName

FK:fName refs fan(fName)

Question assigned to the following page: [3.1](#)



Question assigned to the following page: [3.2](#)

**3.2. (10 points) Exercise 4.2.1 on page 145.** Before you do this problem, read Text book 4.2 Design Principles first from page 140 to page 145.

**Exercise 4.2.1:** In Fig. 4.14 is an E/R diagram for a bank database involving customers and accounts. Since customers may have several accounts, and accounts may be held jointly by several customers, we associate with each customer an "account set," and accounts are members of one or more account sets. Assuming the meaning of the various relationships and attributes are as expected given their names, criticize the design. What design rules are violated? Why? What modifications would you suggest?

Design rules violated:

- Simplicity, the database contain unnecessary entities, which makes the diagram more complex.
- Relationships, some relationships are not necessary, misleading, or not properly connecting the right entities.
- Elements, some elements should be attributes, not entities, and some are unnecessary.

Recommendations:

- I would strongly suggest only having two entities, 'customers' and 'accounts', both linked by the relationship 'customer owns account'.
- The customer entity should have the attributes: name, address, account\_set, account\_number, ...
- The accounts entity should have the attributes: account\_number, account\_type, balance, ...

Question assigned to the following page: [3.3](#)

3.3. (10 points) Exercise 4.4.4 a) on page 156

Exercise 4.4.4: Draw E/R diagrams for the following situations involving weak entity sets. In each case indicate keys for entity sets.

- a) Entity sets Courses and Departments. A course is given by a unique department, but its only attribute is its number. Different departments can offer courses with the same number. Each department has a unique name.

